

HEWLETT-PACKARD

# HP-65

## Kurzanleitung

**Bedienungsanweisungen**

**Tastenzusammenstellung**

**Stichwortverzeichnis**

Diese Broschüre ist hauptsächlich als Nachschlagewerk für Sie gedacht, nachdem Sie das **HP-65 Bedienungshandbuch** gelesen haben. Der Abschnitt **Bedienungsanweisungen** (§ 1–10) ermöglicht Ihnen einen Überblick über die wesentlichen Bedienungsverfahren. Die **Tastenzusammenstellung** (§ 11–67) hilft Ihnen, schnell die Details in Zusammenhang mit den einzelnen Tasten und Schaltern nachzuschlagen. In dieser Zusammenstellung sind alle Symbole des HP-65-Tastenfeldes in alphabetischer Reihenfolge aufgeführt; nicht-alphabetische Symbole (wie **+**, **π**) stehen am Anfang der Aufstellung (§ 11–14). Im Anschluß an die Tastenzusammenstellung finden Sie ein allgemeines **Stichwortverzeichnis**.

## Bedienungsanweisungen (§ 1–10)

### 1. W/PRGM-RUN Schalter

In Stellung **RUN**: Zum Rechnen, um ein Programm auszuführen, zum Lesen einer vorprogrammierten Magnetkarte. • In Stellung **W/PRGM**: Zum Löschen des



Programmspeichers, zum Eintasten eines Programms, zur Korrektur eines Programms, zur Aufzeichnung auf Magnetkarte.

### 2. Arithmetische Operationen im «Stack»

**Berechnen Sie:**      **Drücken Sie:**

$8 - 2 = 6$                       8 **ENTER** 2 **-** → 6

$8 \div 2 = 4$                       8 **ENTER** 2 **÷** → 4

Berechnen Sie:  $\frac{(4 \times 5)}{(2 + 3)} - 6$ , indem Sie die

Tasten in folgender Reihenfolge drücken:

**Registerinhalte**

<b>T</b>											
<b>Z</b>						20	20				
<b>Y</b>		4	4		20	2	2	20		4	
<b>X</b>	4	4	5	20	2	2	3	5	4	6	-2

Tasten **4** **↑** **5** **×** **2** **↑** **3** **+** **÷** **6** **-**

**Beachten Sie:** **ENTER** **↑** ist abgekürzt als **↑**.

### 3. Eingabe von Daten

**Eingabe negativer Zahlen:** Drücken Sie **CHS** (Vorzeichenwechsel = change sign)

**Bedienungsanweisungen**

**Tastenzusammenstellung**

**Stichwortverzeichnis**



nach Eingabe des positiven Wertes. • **Beispiel:** Zur Eingabe von  $-12$  drücken Sie **12 CHS**.

**Eingabe kleiner und großer Zahlen** (Exponentialdarstellung): Methode: ① Tasten Sie die Mantisse ein, ② drücken Sie **CHS** falls die Zahl negativ ist, ③ drücken Sie **EEX**, ④ tasten Sie den Exponent ein, ⑤ drücken Sie **CHS** falls der Exponent negativ ist. • **Beispiel:** Um  $5 \times 10^{-8}$  einzutasten, drücken Sie **5 EEX 8 CHS**.

**Korrektur falscher Eingaben:** Drücken Sie **CLX** und geben Sie den richtigen Wert ein.

## 4. Anzeige

**Wahl der Anzeige:** Um das X-Register in Festkommadarstellung mit  $n$  Nachkommastellen anzuzeigen, drücken Sie **DSP** **[.]** **[n]** ( $n=0, 1, \dots, 9$ ). Um  $x$  in Exponentialdarstellung (wissenschaftl. Schreibweise) mit  $n$  Nachkommastellen anzuzeigen, drücken Sie **DSP** **[n]**. • Die Wahl des Anzeigeformates kann auch von einem Programm aus erfolgen.

**Blinkende Anzeige:** Die Anzeige blinkt bei dem Versuch, eine unerlaubte Operation auszuführen (Beispiel:  $5 \div 0$ ). Das Blinken wird durch Drücken einer beliebigen Taste aufgehoben, ohne daß die dieser



Taste zugeordnete Funktion ausgeführt wird. • Blinken der Anzeige kann auftreten, wenn eine Programmkarte fehlerhaft gelesen wurde (oder leer ist). • Sehen Sie im Abschnitt «Tastenzusammenstellung» bezüglich der Einschränkungen für folgende Operationen nach: **[LN]**, **[LOG]**, **[ $\sqrt{x}$ ]**, **[SIN]**, **[COS]**, **[DMS+]**, **[ $\rightarrow$ DMS]**, **[ $\rightarrow$ OCT]**, **[ $1/x$ ]**, **[ $y^x$ ]**, **[ $\div$ ]**.

**Mehrere Dezimalpunkte:** Das Aufleuchten aller Dezimalpunkte zeigt an, daß die Batteriespannung abfällt.

## 5. Präfix-Operationen

**Goldfarbene Funktionen** (oberhalb der Tasten). Regel: Drücken Sie **[f]** vor der Taste der entsprechenden goldfarbenen Funktion oder aber **[f1]**, wenn Sie die inverse Funktion ausführen möchten. • **Beispiel:** Berechnen Sie  $10^2 = 100$  (inverse Funktion), indem Sie **[f1] [LOG]** drücken.

**Blaue Funktionen** (Tastenunterseite). Drücken Sie vor solchen Funktionen die Taste **[g]**. • **Beispiel:** Berechnen Sie  $5! = 120$ , indem Sie **5 [g] [n!]** drücken.

**Korrektur oder Aufheben eines Präfix:** Zum Korrigieren eines falschen Präfix drücken Sie einfach die Taste des richtigen Präfix. • Zum Aufheben eines Präfix,

**Bedienungsanweisungen**

**Tastenzusammenstellung**

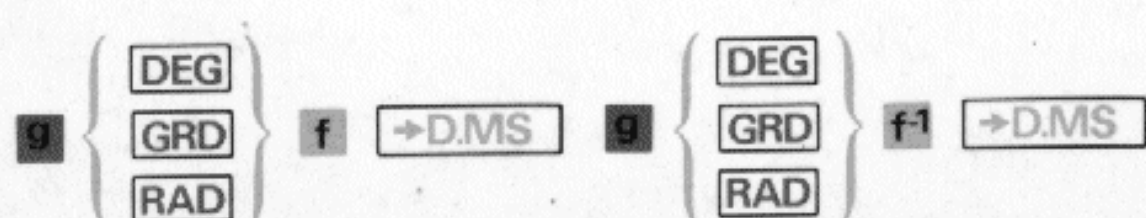
**Stichwortverzeichnis**



drücken Sie **f** **PREFIX**. • Weitere Präfix-Tasten sind **DSP**, **GTO**, **STO**, **RCL**, **LBL**.

## 6. Winkel-Modus

Operationen, die sich auf Winkel beziehen (**→D.MS**, **SIN**, **COS**, **TAN**, **R→P**) erwarten die Winkelargumente in der Einheit (Grad, rad [Bogenmaß], oder Gon [Neugrad]) des geltenden Winkel-Modus, so wie er durch **DEG** (bzw. Einschalten), **RAD**, oder **GRD** eingestellt wurde. •  $360^\circ$  (Grad) =  $2\pi$  rad (Bogenmaß) = 400 g (Neugrad). Umwandlung von einem Winkel-Modus in einen anderen:



## 7. Speicherregister-Operationen

**Speichern einer Zahl in Register  $R_n$ .**

- ① Tasten Sie die zu speichernde Zahl ein,
- ② drücken Sie **STO** **n** (wobei  $n = 1, 2, \dots, 9$ ).

**Rückrufen einer Zahl aus  $R_n$ .** Drücken Sie **RCL** **n** (wobei  $n = 1, 2, \dots, 9$ ).

**Speicherregister-Arithmetik.**

**Subtraktion:**  $(r_n - x \rightarrow R_n)$  **STO** **-** **n**

**Addition:**  $(r_n + x \rightarrow R_n)$  **STO** **+** **n**

**Multiplikation:**  $(r_n \cdot x \rightarrow R_n)$  **STO** **x** **n**

**Division:**  $(r_n \div x \rightarrow R_n)$  **STO** **÷** **n**

wobei **n** eine Zahl von 1, ..., 9 ist. X bleibt bei diesen Operationen unverändert.

**Beachte:**  $R_9$  wird benutzt durch folgende trigonometrische Funktionen und deren Umkehrfunktionen: **SIN** (§ 55), **COS** (§ 19), **TAN** (§ 60), **R→P** (§ 51) und durch die arithmetischen Vergleiche **x≠y**, **x≤y**, **x=y**, **x>y** (§ 66).

## 8. Löschen der Register

Drücken Sie **CLX** um das X-Register zu löschen. • Drücken Sie **f** **STK** um den ganzen «Stack» zu löschen. • Drücken Sie **f** **REG** um alle Speicherregister ( $R_1, \dots, R_9$ ) zu löschen. • Beim Einschalten des Rechners werden sämtliche Register gelöscht.

## 9. Programmierung

**Programmspeicher.** Dient zum Speichern der Benutzerprogramme. Kapazität: 100 Speicherzeilen. • Der Anfang liegt über der ersten Speicherzeile. • Das Ende des Speichers ist die letzte Speicherposition. • Der Programmschritt-Anzeiger, ein interner Teil des Rechners, bestimmt die Speicherzeile, die angezeigt oder ausgeführt werden

Bedienungsanweisungen

Tastenzusammenstellung

Stichwortverzeichnis



soll. Im W/PRGM-Modus werden alle Tastenbefehle in Codeform in den Speicher geschrieben: Der Code für den Speicheranfang ist 00 00. Der Code für die Tasten 0–9 ist 00–09. Für die übrigen Tasten setzt sich der Code aus Zeile und Spalte der Tastenposition auf dem Tastenfeld zusammen. Beispiel: Der Code für **R/S** (Zeile 8, Spalte 4) ist 84.

#### Vom Benutzer festgelegte Funktionen.

Die Tasten in der oberen Reihe werden zum Aufruf benutzereigener Funktionen verwendet. Wenn der Rechner eingeschaltet wird, werden die im Fenster oberhalb der obersten Tastenreihe angegebenen Funktionen in den Programmspeicher geladen. Funktionen nach eigener Wahl können eingetastet oder von Programmkarte in den Speicher gelesen werden. Das Drücken einer der Tasten der oberen Reihe ruft die entsprechende Funktion auf und führt sie aus.

**Schreiben einer Funktion:** Kennzeichnen Sie den Anfang durch **LBL** gefolgt von der Taste der oberen Reihe, über die die Funktion aufgerufen werden soll. Beenden Sie die Funktion mit **RTN**. **Beispiel:** **LBL E** (**ENTER** **ENTER** **x** **x**) **RTN**. Die in Klammern gesetzte Tastenfolge berechnet  $x^3$ .

**Eintasten einer Funktion:** ① W/PRGM-



RUN Schalter in Stellung W/PRGM. ② Drücken Sie **f** **PRGM** um den Programmspeicher zu löschen. ③ Drücken Sie die Tasten in der im Beispiel angegebenen Folge (**LBL E** ... **RTN**). Wenn Sie einen Fehler gemacht haben, drücken Sie **9** **DEL** um den Fehler zu beseitigen, und drücken Sie dann die richtige Taste.

**Ausführen der Funktion** (vom Tastenfeld aus): ① W/PRGM-RUN Schalter in Stellung RUN. ② Drücken Sie die entsprechende Taste der oberen Reihe. Um, wie im Beispiel,  $2^3 = 8$  zu rechnen, drücken Sie **2** **E**, für  $3^3 = 27$  entsprechend **3** **E** usw.

• Eine vom Benutzer geschriebene Funktion kann auch von einem Programm aus ausgeführt werden, indem einfach die Taste der oberen Reihe an gewünschter Stelle in das Programm eingefügt wird.

**Korrektur (Änderung) eines Programms.** Drücken Sie **RTN** im RUN-Mode, um den Programmschritt-Anzeiger an den Anfang des Speichers zu setzen. • Zum Vorrücken des Programmschrittanzeigers auf eine Marke drücken Sie **GTO** **[n]** im RUN-Mode, wobei n die gleiche Zahl oder Taste der oberen Reihe darstellt wie bei **LBL** **[n]** (Marke n). • Mit der Taste **SST**

**Bedienungsanweisungen**

**Tastenzusammenstellung**

**Stichwortverzeichnis**



im W/PRGM-Mode können Sie schrittweise die Verschlüsselungen der einzelnen Programmbefehle anzeigen (siehe § 56). Um nach dem augenblicklich angezeigten Programmschritt einen neuen Programmbefehl einzusetzen, brauchen Sie lediglich den neuen Schritt im W/PRGM-Mode einzutasten (siehe § 32). • Zum Löschen eines Programmschrittes oder zur Beseitigung eines Fehlers, drücken Sie **g** **DEL** im W/PRGM-Mode (siehe § 21). Ein laufendes Programm wird durch Über- oder Unterlauf in einem Register gestoppt.

## 10. Verwendung von Magnetkarten

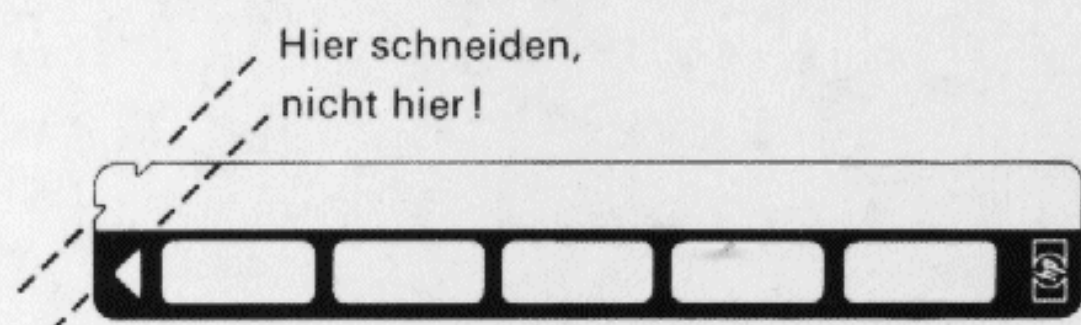
**Lesen einer vorprogrammierten Karte.** W/PRGM-RUN Schalter in Stellung RUN.

• Führen Sie die Karte in den rechten unteren Schlitz ein. • Falls die Programmkarte nicht fehlerfrei gelesen wurde, blinkt die Anzeige und der Programmspeicher wird gelöscht, ohne Effekt auf die Registerinhalte. Drücken Sie **R/S** und führen Sie die Karte erneut ein.

**Aufzeichnen (Schreiben) auf eine Programmkarte.** W/PRGM-RUN Schalter in Stellung W/PRGM. • Führen Sie eine ungeschützte Karte (ohne Eckenabschnitt) in den rechten unteren Schlitz ein. • Schützen



Sie die Programmkarte, indem Sie den markierten Eckenabschnitt entfernen.



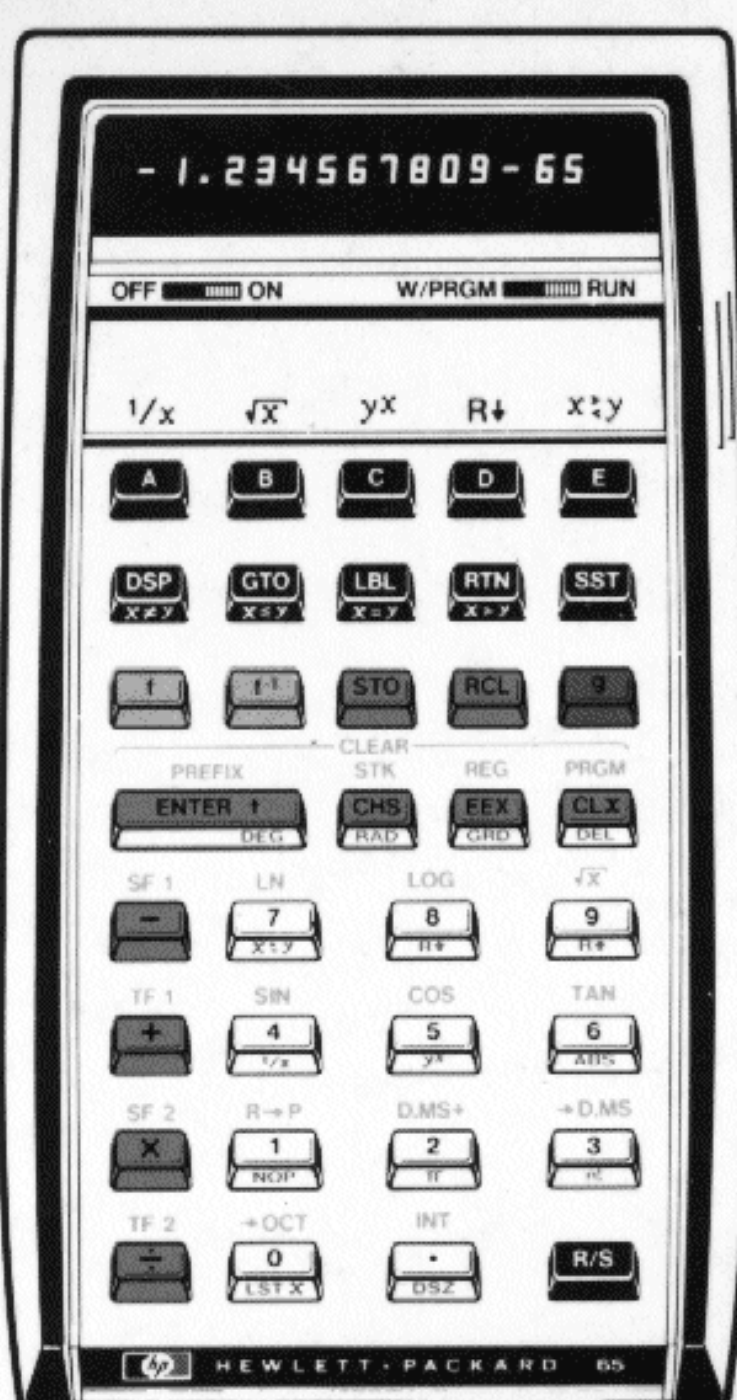
## Tastenzusammenstellung (§ 11–67)

Dieser Abschnitt enthält Referenzvermerke für alle Symbole, die auf dem HP-65-Tastenfeld auftreten. Zusätzlich sind folgende Begriffe mit aufgenommen: **Insert**, § 32 – **Last X**, § 34 – **Kombinierte Codes**, § 39 – **Ein/Aus Schalter**, § 43 – **Stack-Lift**, § 57 – **W/PRGM-RUN Schalter**, § 62.

## Tastenzusammenstellung

### Stichwortverzeichnis





Klaviatur HP-65

## 11. $-$ , $+$ , $\times$ , $\div$ Arithmetische Operationen

Berechnen jeweils  $(y-x)$ ,  $(y+x)$ ,  $(y \cdot x)$ ,  $(y \div x)$ . • Ermöglicht Stack-Lift. • Speichert  $x$  nach Last X. • Stack fällt wie folgt:



Inhalt der Register	Name der Register
t	T
z	Z
y	Y
x	ergibt X
	Last X

Wegen Speicherregister-Arithmetik siehe § 59.

## 12. $\square$ ... $\square$ Ziffern

Ziffern werden verwendet: ① zur Eingabe von Daten (§ 3), ② zur Bezeichnung der Register (§ 59 und 49), ③ zum Markieren von Programmschritten (§ 35 und 31), ④ zur Angabe der anzuzeigenden Dezimalstellen (§ 24). • Werden die Zifferntasten zur Eingabe von Zahlen gedrückt, ermöglichen sie den «Stack-Lift» (§ 57).

## 13. $\square$ Dezimalpunkt

Die  $\square$ -Taste wird auch zur Wahl der Festkommaanzeige genutzt (§ 24). • Ermöglicht «Stack-Lift».

## 14. $\pi$ Gibt $\pi$ ( $= 3,141592654$ ) in das X-Register des Stack ein

Hebt den Stack falls «Stack-Lift» ermöglicht. • Ermöglicht «Stack-Lift».

## 15. $\mathbf{A}$ , $\mathbf{B}$ , $\mathbf{C}$ , $\mathbf{D}$ , $\mathbf{E}$ Tasten der oberen Reihe

Ohne Präfix gedrückt, ruft die Taste der



oberen Reihe die ihr entsprechende benutzereigene Funktion auf und führt sie aus.

- **Zu beachten:** Beim Einschalten des Rechners werden unter **A**, ..., **E** automatisch fünf Unterprogramme in den Speicher geladen.
- Drücken Sie **f** **PRGM** im W/PRGM-Modus um sie zu löschen.
- Die Tasten der oberen Reihe ändern den «Stack-Lift»-Status nicht.
- Die Tasten werden auch im Zusammenhang mit **LBL** und **GTO** verwendet (§ 35 bzw. 31).

## 16. **ABS** Absolutwert ( $|x|$ )

**ABS** macht den negativen Inhalt des X-Registers positiv.

- Speichert x nach Last X.
- Ermöglicht «Stack-Lift».

**B** und **C** siehe § 15.

## 17. **CHS** Vorzeichenwechsel (change sign)

Ändert das Vorzeichen des Inhalts im X-Register, solange die **EEX**-Taste noch nicht gedrückt wurde; anderenfalls wird das Vorzeichen des Exponenten umgekehrt.

- Die erste von Null verschiedene Ziffer einer Zahleneingabe setzt das X-Register positiv.
- Wird die Zahleneingabe beendet (durch eine andere Taste als eine Zifferntaste, **CHS** oder **EEX**), dann kann das Vorzeichen des Exponenten mit **CHS** nicht mehr geändert werden. In diesem Fall wechselt bei Drücken von **CHS** das Vorzeichen der Zahl.
- Kein Einfluß auf «Stack-Lift».

## 18. **CLX** Löscht X-Register

Verhindert «Stack-Lift».

## 19. **COS** Kosinus (Präfix **f**)/Arkuskosinus (Präfix **f<sup>-1</sup>**)

(Hauptwert,  $0^\circ \leq \text{Resultat} \leq 180^\circ$  oder entsprechend in rad [Bogenmaß] bzw. Gon [Neugrad]).

- Speichert x nach Last X.
- Löscht Inhalt von Register R<sub>9</sub>.
- X kleiner -1 oder größer +1 führt bei arccos zu Fehleranzeige (Blinkende Null).
- Ermöglicht «Stack-Lift».

**D** Siehe § 15.

## 20. **DEG** Setzt Winkel-Modus auf Grad

Bezieht sich auf Winkeloperationen (**→DMS**, **SIN**, **COS**, **TAN**, **R→P**).

- Kein Einfluß auf «Stack-Lift».

## 21. **DEL** Löscht Programmschritt

Bei W/PRGM-RUN Schalter in Stellung W/PRGM, bewirkt Drücken von **g** **DEL**:

- ① Der vom Programmschritt-Anzeiger angezeigte Programmschritt (§ 9) wird gelöscht,
- ② alle nachfolgenden Programmschritte werden um eine Speicherzeile nach oben gerückt,
- ③ ein **g** **NOP**-Code wird in die freigewordene letzte Speicherposition geschrieben.

- Nicht wirksam im RUN-Modus, wirkt in solchem Fall wie **CLX**.
- **g** **DEL** kann zum Zurückstellen des Programmschritt-Anzeigers verwendet werden



(worauf Sie den korrigierten Programmschritt eingeben).

**Falls der Programmspeicher voll ist** (zu sehen am Minuszeichen in der Anzeige), geht beim Drücken von **g** **DEL** der Programmschritt in der letzten Speicherzeile verloren.

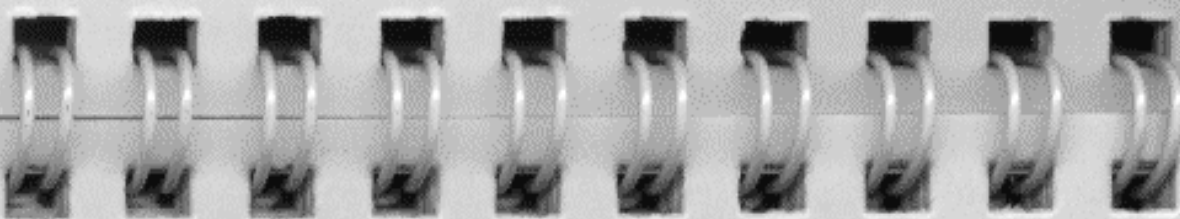
**Steht der Programmschritt-Anzeiger am Ende des Speichers** (zwei Minuszeichen in der Anzeige), löscht **g** **DEL** zwei Speicherzeilen. • Kein Einfluß auf «Stack-Lift».

**22. **→D.MS** Wandelt nach (Präfix **f**) oder von (Präfix **f1**) Grad, Minuten, Sekunden um**

Wandelt den Inhalt des X-Registers (in Grad, rad oder Neugrad) in die Form DDDDD.MMSS (Grad, Minuten, Sekunden) um, oder umgekehrt, je nach Präfix. • Speichert x nach Last X. • Führt zu Fehleranzeige (blinkende Null), falls der Betrag im X-Register (in Grad oder entsprechend in rad oder Neugrad) den Wert 99999.99999 übersteigt. • Ermöglicht «Stack-Lift».

**23. **D.MS+** Addiert (Präfix **f**) / subtrahiert (Präfix **f1**) Grad, Minuten, Sekunden**

**D.MS+** addiert ( $y+x$ ) bzw. subtrahiert ( $y-x$ ) die in der Form DDDDD.MMSS im X- und Y-Register stehenden Winkel. • Speichert x nach Last X. • Stack fällt (wie in §11). • Führt zu Fehleranzeige (blin-



kende Anzeige 0.00), wenn einer der Operanden oder das Resultat den Wert 99999.59599 Grad, Minuten und Sekunden übersteigt. • Ermöglicht «Stack-Lift».

**24. **DSP** Anzeige (Präfix)**

**DSP** **•** **n** zeigt x auf n Nachkommastellen gerundet an. • **DSP** **n** schaltet auf Exponentialdarstellung (wissenschaftliche Schreibweise), gerundet auf n Nachkommastellen, mit  $n=0, 1, \dots, 9$ . • Ändert nicht den internen Wert von X. Einschalten des Rechners wirkt wie **DSP** **•** **2** (0.00). • Ist x für ein gewähltes Anzeigeformat zu klein, wird Null angezeigt. • Ist x für das entsprechende Format zu groß, schaltet der Rechner auf Exponentialdarstellung: **DSP** **9**. • Kein Einfluß auf «Stack-Lift».

**25. **DSZ** Dekrement und Sprung bei Null**

Subtrahiert 1 von einer ganzen Zahl in  $R_8$ ; führt zum Überspringen von zwei Programmschritten, falls  $r_8=0$ . • Die Dekrementfunktion wird außerhalb der folgenden Grenzen unterdrückt:  $1 \leq r_8 \leq 10^{10}$ . • Nützlich für das Bilden von Programmschleifen. Regel: Um ein markiertes Programmsegment n-mal auszuführen, speichere n nach  $R_8$  und nutze **DSZ** um zu entscheiden, ob der Programmteil wiederholt werden soll oder nicht. **Beispiel:** Die



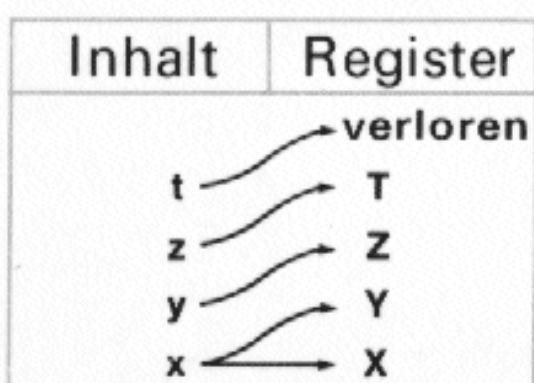
folgende Schleife wird 12 mal durchlaufen, dann hält das Programm an: 12 **STO** **8** **LBL** **1** ... **9** **DSZ** **GTO** **1** **R/S**. • Kein Einfluß auf «Stack-Lift».

**E** Siehe § 15.

## 26. **EEX** Eingabe Exponent

Beendet die Eingabe der Mantisse einer Zahl und zeigt die Eingabe eines Exponenten (zur Basis 10) in das X-Register an. Wurde vorher keine Mantisse eingetastet, setzt **EEX** die Mantisse automatisch gleich 1. • Ermöglicht «Stack-Lift».

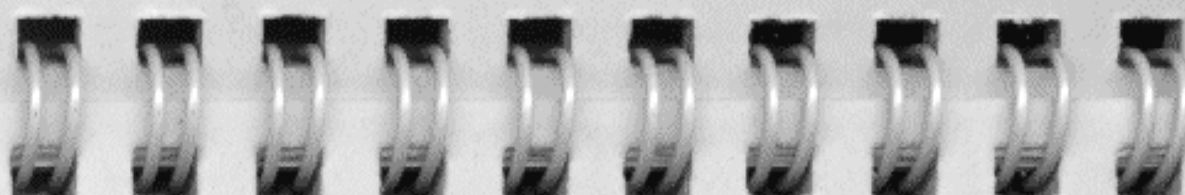
## 27. **ENTER** Übertrage x nach Y



• Kopiert den Inhalt des X-Registers in das Y-Register, schreibt y nach Z und z nach T (t geht verloren). • Verhindert «Stack-Lift».

## 28. **f** **f<sup>-1</sup>** Präfix für Funktionen oberhalb der Tasten (goldfarbene Funktionen)

Das goldfarbene Symbol über einer Taste bezeichnet die Funktion der Taste, falls vorher **f** gedrückt wurde. • Die inverse Funktion bzw. das Komplement ergibt sich, wenn vorher **f<sup>-1</sup>** gedrückt wurde. • Für die Tasten in der vierten Zeile (Lösch-Funktionen) sind keine entsprechenden inversen



Funktionen definiert. Hier bewirkt auch **f<sup>-1</sup>** die jeweilige Löschfunktion. • Zum Aufheben eines unerwünschten **f** oder **f<sup>-1</sup>**, drücken Sie **PREFIX**. • Kein Einfluß auf «Stack-Lift».

## 29. **g** Präfix für Funktionen auf der Tastenunterseite (blaue Funktionen)

Wird im Anschluß an **g** eine Funktions-taste gedrückt, so wird die auf der Tastenunterseite blau markierte Funktion ausgeführt. • Zum Aufheben eines unerwünschten **g**, drücken Sie **f** **PREFIX**. • Kein Einfluß auf «Stack-Lift».

## 30. **GRD** Stellt Winkel-Modus auf Neugrad

Bezieht sich auf Winkeloperationen (**→DMS**, **SIN**, **COS**, **TAN**, **R→P**). • Kein Einfluß auf «Stack-Lift».

## 31. **GTO** Gehe nach... (Go to) (Präfix)

Sprungbefehl; folgt auf ihn eine Ziffer (**0**, ..., **9**) oder einer der Buchstaben (**A**, ..., **E**), dann wird der Programmschritt-Anzeiger bis zum Erreichen der entsprechenden Marke (**LBL**, gefolgt von der gleichen Ziffer oder gleichem Buchstaben) vorgerückt. • Kein Einfluß auf «Stack-Lift».

## 32. Insert

Wird im W/PRGM-Modus eine Taste ge-



drückt, so wird sie kodiert und dieser Code zwischen den angezeigten und den nachfolgenden Programmschritt in den Programmspeicher geladen. Der Programmschritt-Anzeiger wird um eine Zeile weitergerückt und zeigt jetzt den eingefügten Programmschritt an. Dabei wird der letzte Programmschritt (Speicherende) aus dem Speicher geschoben. • Kein Einschieben (Insert) findet statt: für **PRGM**, **DEL**, **SST**, für die zweite Taste eines kombinierten Codes, falls der Programmschritt-Anzeiger am Speicherende steht. • Kein Einfluß auf «Stack-Lift».

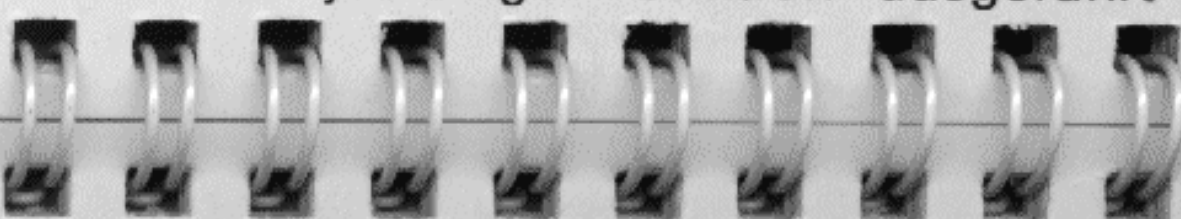
**33. **INT** Macht x ganzzahlig** (streicht Dezimalteil von x) mit Präfix **f** oder **streicht ganzzahligen Anteil von x** mit Präfix **f-1**

Speichert x nach Last X. • Behält das Vorzeichen der Zahl bei. • Ermöglicht «Stack-Lift».

### **34. Last X-Register**

Enthält den letzten Wert von x (bevor das letzte  $f(x)$  bzw.  $f(x \cdot y)$  berechnet wurde).

• Wird genutzt für: • Berechnung von Funktionen, die wiederholt den gleichen Operanden verwenden. • Fehlerbeseitigung beim Drücken falscher Tasten in manchen Fällen. • Für das Zurückrufen des letzten X-Wertes siehe **LST X** (§ 38). Die folgenden Tasten speichern x nach Last X, bevor die jeweilige Funktion ausgeführt



wird: **+**, **-**, **x**, **÷**, **ABS**, **COS**, **→D.MS**, **D.MS+**, **INT**, **LN**, **LOG**, **→OCT**, **R→P**, **SIN**, **TAN**, **n!**, **√x**, **1/x**, **y<sup>x</sup>**. • Beachten Sie, daß sich **CLX** nicht auf das Last - X-Register bezieht.

### **35. **LBL** Marke (Label) (Präfix)**

**LBL** kennzeichnet die nachfolgende Ziffer (**0**, ..., **9**) oder den nachfolgenden Buchstaben (**A**, ..., **E**) als Marke in einem gespeicherten Programm. • Ein Sprung zu einem solchermaßen markierten Programmteil kann dann durch eine Sprunganweisung der Form **GTO**, gefolgt vom gleichen Suffix, bewirkt werden. • Für vom Benutzer geschriebene Funktionen muß der auf **LBL** folgende Suffix eine der Buchstabentasten der oberen Reihe sein (**A**, ..., **E**). • Kein Einfluß auf «Stack-Lift».

### **36. **LN** Natürlicher Logarithmus (Präfix **f**) oder **e<sup>x</sup>** (Präfix **f-1**)**

Speichert x nach Last X. • **f** **LN** ergibt Fehlermeldung (blinkende Anzeige) falls x gleich oder kleiner als Null. • Ermöglicht «Stack-Lift».

### **37. **LOG** Log<sub>10</sub> (Präfix **f**) oder **10<sup>x</sup>** (Präfix **f-1**)**

Speichert x nach Last X. • **f** **LOG** ergibt Fehlermeldung (blinkende Anzeige) falls x gleich oder kleiner als Null. • Ermöglicht «Stack-Lift».



### 38. **LST X** Ruft Inhalt von Last X in X-Register zurück

Führt selbständig «Stack-Lift» durch, falls «Stack-Lift» nicht verhindert wurde. • Ermöglicht «Stack-Lift».

### 39. Kombinierte Codes

Der Code der folgenden Tasten ist mit dem Code der entsprechenden Folgetasten (Suffix) kombiniert: **LST X**, **NOP**, **x $\nabla$ y**, **R $\downarrow$** , **R $\uparrow$** , **x $\neq$ y**, **x $\leq$ y**, **x=y**, **x $>$ y**, **1**, ..., **8** falls **STO** oder **RCL** vorangegangen. • Beispiel: **9** **LST X** wird im Programm-Modus kombiniert und als 35 00 angezeigt; **STO** **5** als 33 05 usw. Erklärung des Codes siehe § 9.

### 40. **n!** n-Fakultät

Berechnet die Fakultät einer nicht-negativen ganzen Zahl n im X-Register.  $n! = 1 \cdot 2 \cdot \dots \cdot (n-1) \cdot n$ .  $0! = 1$  (nach Definition). • Speichert x nach Last X. • Negative oder nicht ganzzahlige x führen zu Fehlermeldung (blinkende Null). • Falls x größer ist als 69, tritt Überlauf auf. • Ermöglicht «Stack-Lift».

### 41. **NOP** Leerbefehl (keine Funktion)

Nützlich zum Auffüllen von Speicherbereichen bei Tests. • Beim Löschen über **f** **PRGM** im W/PRGM-Modus wird der gesamte Programmspeicher mit **9** **NOP** (kombinierter Code 35 01) beschrieben. • Kein Einfluß auf «Stack-Lift».



### 42. **→OCT** Wandelt ganzzahlige x nach/von Oktal-Darstellung um

Speichert x nach Last X. • Führt zu Fehleranzeige (blinkende Null), falls x nicht ganzzahlig oder größer ist als  $1073741823_{10}$ . • Ermöglicht «Stack-Lift».

### 43. Ein/Aus-Schalter

«Einschalten»: Löscht alle Register und Flags. • Stellt Winkel-Modus auf Grad. • Stellt die Anzeige auf Festkommadarstellung mit 2 Nachkommastellen (0.00). • Setzt den Programmschritt-Anzeiger an den Speicheranfang. Lädt fünf Funktionen in den Programmspeicheranfang, die mit einem Tastendruck (Tasten der oberen Reihe) ausgeführt werden können. Den Funktionen entsprechen die in dem Fenster über der oberen Tastenreihe stehenden Symbole.

### 44. **PREFIX** Lösche Präfix

Im Anschluß an **f** oder **f1** gedrückt hebt **PREFIX** die Wirkung eines bereits eingetasteten Präfix auf, sodaß eine Einzeltasten-Operation ausgeführt werden kann. • Wurde zu einer Funktionstaste der falsche Präfix eingetastet, die Folgetaste aber noch nicht gedrückt, so genügt es, die richtige Präfixtaste zu drücken und von da fortzufahren. • Kein Einfluß auf «Stack-Lift».

### 45. **PRGM** Lösche Programmspeicher

Im W/PRGM-Modus löscht **f** **PRGM**

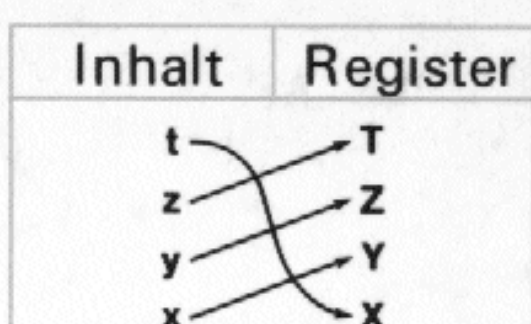
## Tastenzusammenstellung

## Stichwortverzeichnis



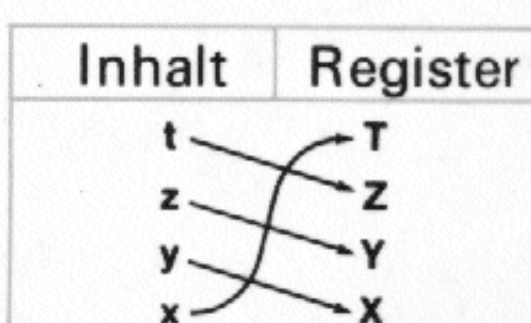
(oder **f1** **PRGM**) den gesamten Programmspeicher und belegt ihn mit **g** **NOP**-Codes (35 01); der Programmschritt-Anzeiger bleibt am Speicheranfang. • Im RUN-Modus ist **f** **PRGM** gleichbedeutend mit **CLX**. • Kein Einfluß auf «Stack-Lift».

#### 46. **R↑** Zyklisches Vertauschen der Register nach oben



Ermöglicht «Stack-Lift».

#### 47. **R↓** Zyklisches Vertauschen der Register nach unten



Ermöglicht «Stack-Lift».

#### 48. **RAD** Stellt Winkel-Modus auf Bogenmaß (rad)

Betrifft Winkeloperationen (**→DMS**, **SIN**, **COS**, **TAN**, **R→P**). • Kein Einfluß auf «Stack-Lift».

#### 49. **RCL** Rückruf (Präfix)

Ruft den Inhalt der Speicherregister  $R_n$  ( $n = 1, \dots, 9$ ) in das X-Register. • Hebt den Stack sofern nicht «Stack-Lift» verhindert ist. • Ermöglicht «Stack-Lift».

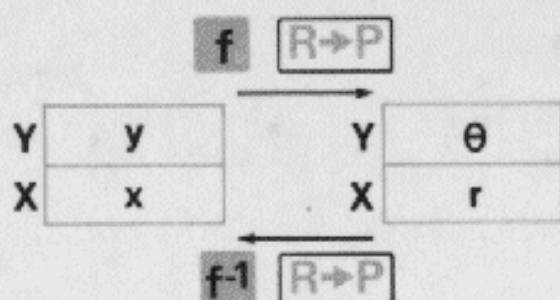


#### 50. **REG** Löscht Konstantenregister

Im Anschluß an **f** oder **f1** gedrückt löscht **REG** die Speicherregister  $R_n$  ( $n = 1, \dots, 9$ ) zu Null.

#### 51. **R→P** Umwandlung rechtwinkliger Koordinaten in Polarkoordinaten

Wandelt die Inhalte des X- und Y-Registers von der rechtwinkligen (kartesischen) Koordinatendarstellung in Polarkoordinaten um  $(x, y) \rightarrow (r, \pi)$  oder umgekehrt, in Abhängigkeit des Präfix (**f** oder **f1**). • Speichert  $x$  nach Last X. • Zerstört den bisherigen Inhalt von Register  $R_9$ . • Ermöglicht «Stack-Lift».



#### 52. **R/S** Start/Stop

Wenn **R/S** gedrückt wird und ein gespeichertes Programm im Augenblick nicht läuft, startet **R/S** die Programmausführung ab der Stelle, an der sich der Programmschritt-Anzeiger gerade befindet. • Im Rahmen eines Programms ausgeführt, führt **R/S** zum Anhalten des Programms und ermöglicht dem Benutzer die Eingabe von Werten oder das Drücken anderer Tasten. In diesem Fall bleibt der Programmschritt-



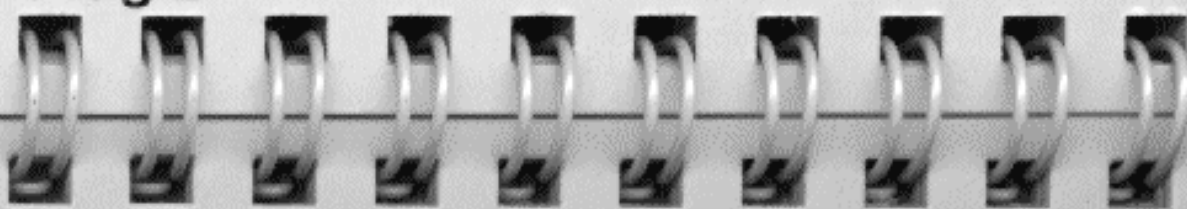
Anzeiger an der Stelle des **R/S**-Befehls stehen. • Geht dem **R/S**-Befehl in einem Programm unmittelbar die Eingabe von Zahlen durch das Programm voraus, wird der automatische «Stack-Lift» bei Übergabe der Kontrolle an das Tastenfeld verhindert. Das ermöglicht dem Programm die Anzeige von Zwischenergebnissen, die bei einer weiteren Dateneingabe durch den Benutzer nicht im Stack angehoben sondern gelöscht (überschrieben) werden. Mit dieser Einschränkung beeinflusst **R/S** den «Stack-Lift» nicht.

### 53. **RTN** Zurück! (Return!)

Wird **RTN** vom Tastenfeld aus gedrückt, setzt es den Programmschritt-Anzeiger an den Speicheranfang. • In einem gespeicherten Programm ist **RTN** das logische Ende einer vom Benutzer definierten Funktion. Wird eine Funktion vom Tastenfeld aus aufgerufen, stoppt **RTN** das Programm.

• Wird die Funktion von einem Programm aus aufgerufen, geht mit der Ausführung von **RTN** die Kontrolle an das Hauptprogramm zurück. • Eine vom Tastenfeld aus aufgerufene Funktion oder ein Hauptprogramm können ihrerseits eine Funktion aufrufen; diese kann dann allerdings nicht ihrerseits eine weitere Funktion aufrufen. • Kein Einfluß auf «Stack-Lift».

### 54. **SF1**, **SF2** Setze Flag 1, setze Flag 2



**f** **SF1** setzt Flag 1 während **f-1** **SF1** Flag 1 löscht. • Entsprechendes gilt für **SF2** bezüglich Flag 2. Zum Testen der Flags siehe § 61. • Kein Einfluß auf «Stack-Lift».

### 55. **SIN** Sinus (Präfix **f**)/Arkussinus (Präfix **f-1**)

Arcsin ( $= \sin^{-1}$ ) berechnet den Hauptwert ( $-90^\circ \leq \text{Resultat} \leq +90^\circ$  oder entsprechend in rad oder Gon = Neugrad). • Speichert x nach Last X. • Zerstört Inhalt von Register R<sub>9</sub>. • Arcsin von x mit x kleiner als -1 oder größer als +1 führt zu Fehlermeldung (blinkende Anzeige 0.00). • Ermöglicht «Stack-Lift».

### 56. **SST** Einzelschritt

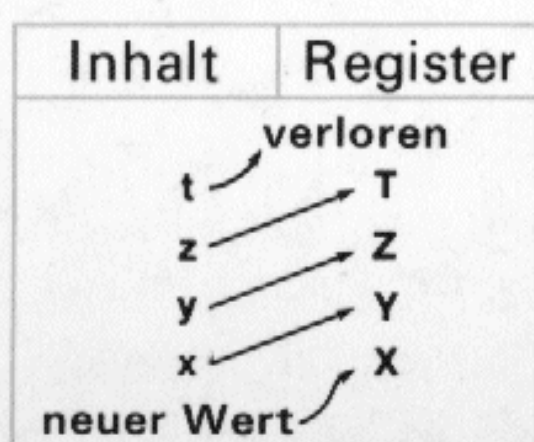
**SST** rückt im W/PRGM-Modus den Programmschritt-Anzeiger auf die nächste Speicherzeile vor und zeigt den Programmschritt als Code an. • Wiederholtes Drücken von **SST** ermöglicht schrittweises Kontrollieren des Programms und positioniert den Programmschritt-Anzeiger für Korrekturzwecke. Gelangt man bei diesem schrittweisen Vorgehen zu dem Aufruf einer vom Benutzer definierten Funktion, dann wird die gesamte Funktion abgearbeitet (als ein Schritt), bevor die Kontrolle an das Tastenfeld zurückgegeben wird. • Kein unmittelbarer Einfluß auf «Stack-Lift».



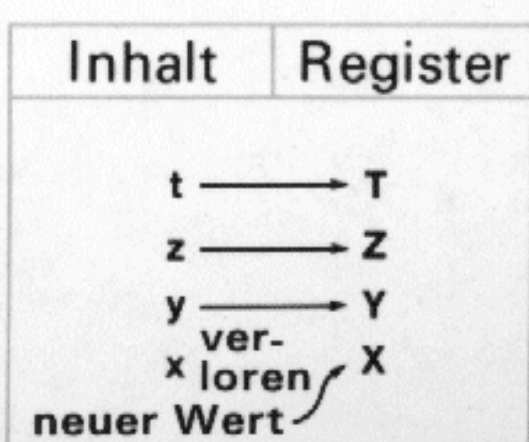
## 57. «Stack-Lift»

Der «Stack-Lift» (Heben des Rechenregister-Stapels) ist eine automatische Einrichtung des HP-65, der Ihnen die Eingabe eines neuen Wertes in das X-Register ermöglicht und dabei gleichzeitig die vorherigen Werte x, y und z in die entsprechend nächsthöheren Register schiebt (t geht dabei verloren). Dies erspart in vielen Fällen die Verwendung der **ENTER↑**-Taste. Ist der «Stack-Lift» ermöglicht, findet er vor Drücken einer der folgenden Tasten wie nebenstehend gezeigt statt: **Π**, **LST X**, **RCL** **[n]** und vor Eingabe der ersten Ziffer einer Zahl in das X-Register. Ist der «Stack-Lift» verhindert, geht der letzte Wert von X verloren (d. h. wird überschrieben).

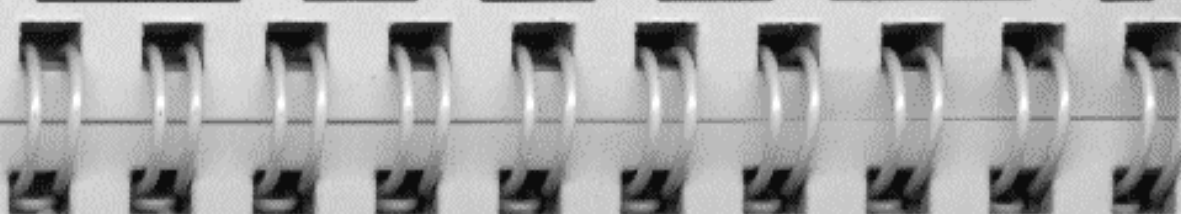
### «Stack-Lift» ermöglicht



### «Stack-Lift» verhindert



«Stack-Lift» wird ermöglicht von: allen Zahleneingabe-Tasten **[0]**, ..., **[9]**, **[.]**, **EEX**, **Π**, aber nicht von **CHS**; allen Rechen-Tasten **−**, **+**, **×**, **÷**, **ABS**, **COS**, **→D.MS**, **D.MS+**, **INT**, **LN**, **LOG**, **LST X**, **[n!]**,



**→OCT**, **R→P**, **SIN**, **TAN**, **1/x**, **√x**, **yx**; auf den Stack bezogene Tasten **R↑**, **R↓**, **x↔y**, aber nicht von **ENTER↑**. • Speicherregister-Tasten **STO**, **RCL**.

**Keinen Einfluß haben:** sämtliche Programmier-Tasten, Winkel-Modus-Tasten, Anzeigekontrolltasten, Lösch-Tasten (außer **CLX**) und **CHS**.

## 58. **[STK]** Lösche Stack

(Rechenregister-Stapel), Präfix **f** oder **f1**.

## 59. **STO** Speicherbefehl (Präfix)

**STO** **[n]** kopiert den Inhalt des X-Registers in das Speicherregister  $R_n$  ( $n = 1, \dots, 9$ ). Wollen Sie mit dem Register  $R_n$  eine arithmetische Operation ausführen, fügen Sie die entsprechende Arithmetik-Taste zwischen **STO** und **[n]** ein.

<b>STO</b> {	<b>+</b>	addiert	$(r_n + x) \rightarrow R_n$
	<b>−</b>	subtrahiert	$(r_n - x) \rightarrow R_n$
	<b>×</b>	multipliziert	$(r_n \times x) \rightarrow R_n$
	<b>÷</b>	dividiert	$(r_n \div x) \rightarrow R_n$

Die Registerinhalte des Stack und Last X werden nicht geändert. • Speicherregister-Arithmetik-Codes sind nicht kombiniert. • Ermöglicht «Stack-Lift».

## 60. **[TAN]** Tangens (Präfix **f**)/Arkustangens (Präfix **f1**)

Arctan x ermittelt den Hauptwert ( $-90^\circ \leq \text{Resultat} \leq +90^\circ$  oder entsprechend in rad



[Bogenmaß] bzw. Gon [Neugrad]). • Speichert x nach Last X. • Zerstört Inhalt von Register R<sub>9</sub>. • Ermöglicht «Stack-Lift».

**61.  $\boxed{\text{TF1}}$ ,  $\boxed{\text{TF2}}$  Teste Flag 1, teste Flag 2**  
**f**  $\boxed{\text{TF1}}$  testet Flag 1 und überspringt zwei Programmzeilen, falls Flag 1 nicht gesetzt ist. **f-1**  $\boxed{\text{TF1}}$  springt zwei Programmschritte weiter, falls Flag 1 gesetzt ist. • Gleiches gilt für **f**  $\boxed{\text{TF2}}$  bzw. **f-1**  $\boxed{\text{TF2}}$ , nur daß es sich auf Flag 2 bezieht. • Zum Setzen der Flags, siehe § 54. • Kein Einfluß auf «Stack-Lift».

## 62. W/PRGM-RUN Schalter

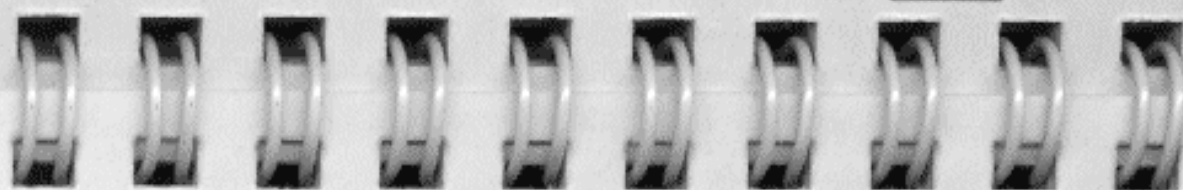
Position W/PRGM schaltet in Programm-Modus für: • Erstellen und Redigieren eines gespeicherten Programms. • Aufzeichnen des Programmspeichers auf Magnetkarte. • Position RUN schaltet in RUN-Modus für: • Einlesen einer Magnetkarte in den Programmspeicher, • Ausführung von Rechnungen, • Ausführen gespeicherter Programme.

## 63. $\boxed{1/x}$ Reziprokwert von x

Speichert x in Last X. • Führt zu Fehlermeldung (blinkende Null), falls  $x = 0$ . • Ermöglicht «Stack-Lift».

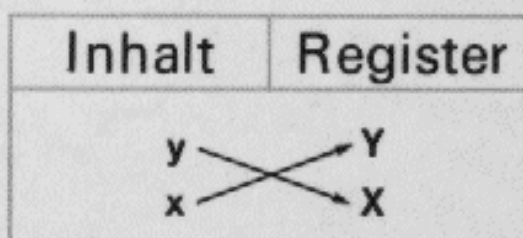
## 64. $\boxed{\sqrt{x}}$ Quadratwurzel von x (Präfix **f**)/ $x^2$ (Präfix **f-1**)

Speichert x in Last X. • Für **f**  $\boxed{\sqrt{x}}$  führt



negatives x zu Fehlermeldung (blinkende Null). • Ermöglicht «Stack-Lift».

## 65. $\boxed{x \leftrightarrow y}$ Tausche x und y aus



Ermöglicht «Stack-Lift».

## 66. $\boxed{x \neq y}$ , $\boxed{x \leq y}$ , $\boxed{x = y}$ , $\boxed{x > y}$ Verhältnis-Tests von x und y

Jeder Test vergleicht den Wert im X- und Y-Register und überspringt zwei Programmzeilen falls die Testbedingung nicht erfüllt ist. • Zerstört Inhalt von Register R<sub>9</sub>. • Kein Effekt auf «Stack-Lift».

## 67. $\boxed{y^x}$ Exponentialfunktion

Berechnet den Ausdruck  $y^x$ . • Speichert x in Last X. • Schiebt Stack nach unten (wie in § 11). • Für y gleich oder kleiner als Null tritt Fehleranzeige auf (blinkende Null). • Ermöglicht «Stack-Lift».



# Stichwortverzeichnis

Einzelheiten über die Tastenfeld-Symbole entnehmen Sie bitte der vorangehenden «Tastenzusammenstellung».

## A

---

Absolutwert, § 16

Addition

Stack (Rechenregister-Stapel), § 11, § 2

Grad, Minuten, Sekunden, § 23

Speicherregister, § 59

Anhalten eines Programms, § 52

Anzeige

Festkomma und Exponentialdarstellung, § 24

blinkende Anzeige, § 4

im W/PRGM-Modus (Befehlcode), § 9

Arithmetische Operationen

im Stack, § 11

mit den Speicherregistern, § 59, § 7

Arkuskosinus, § 19, § 6

Arkussinus, § 55, § 6

Arkustangens, § 60, § 6

## B

---

Blinkende Anzeige, § 4

Bogenmaß (Winkelmodus), § 48, § 6

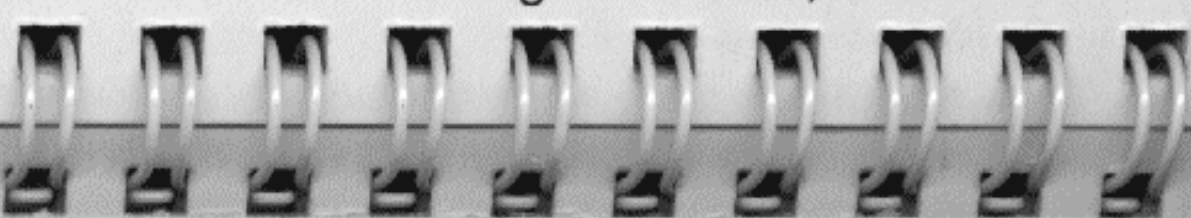
## D

---

Dezimal nach Oktal, § 42

Dezimalpunkt

Wahl des Anzeigeformates, § 24



Eingabe von Zahlen, § 13

Dezimalteil einer Zahl, § 33

## E

---

$e^x$ , § 36

Ein/Aus-Schalter, § 43

Eingabe des Exponenten, § 26

Einschieben eines Programmschrittes, § 32

Einzelschritt, § 56, § 9

Exponentialdarstellung

Anzeige, § 24, § 4

Eingabe von Zahlen, § 3, § 26, § 17

Exponentialfunktion

$10^x$ , § 37

$e^x$ , § 36

## F

---

Fakultät einer Zahl ( $n!$ ), § 40

Fehler, § 4

Flags

Setzen, § 54

Löschen, § 61

Format der Anzeige, § 24, § 4

## G

---

Ganzzahliger Teil von  $x$ , § 33

Gleichheit (Test), § 66

Gon (Neugrad), Winkelmodus, § 30, § 6

Grad, Minuten, Sekunden

Addition, § 23

Umwandlung in Grad dezimal, § 22, § 6

Grad (Winkelmodus), § 20, § 6

größer als (Test), § 66



## H

---

Hebt den Stack, § 57

## I

---

Inverse Operationen

$10^x$ , § 37

Arkuskosinus, § 19

Arkussinus, § 55

Arkustangens, § 60

Grad, Minuten, Sekunden nach Winkel  
dezimal, § 22

$e^x$ , § 36

Dezimalteil einer Zahl, § 33

oktal nach dezimal, § 42

polar nach rechtwinklig, § 51

Flags löschen, § 54

Quadrat, § 64

subtrahiere D.MS, § 23

teste Flags auf «nicht-gesetzt», § 61

## K

---

Keine Operation (NOP), § 41

Code

Erklärung, § 9

kombinierte Codes, § 39

Kosinus, § 19, § 6

## L

---

Last X, § 34

Logarithmus

zur Basis 10, § 37

natürlich, zur Basis  $e$ , § 36

Löschen

Flags, § 54

Programmschritt, § 21, § 9

Programmspeicher, § 45

Register, § 50

Stack, § 58

ganzer Rechner, § 43

X-Register, § 18

## M

---

Magnetkarte, § 10

Marke, § 35, § 9

Multiplikation

im Stack, § 11, § 2

der Speicherregister, § 59

## N

---

$n!$ , § 40

negative Zahlen, § 17, § 3

Neugrad (Gon), Winkelmodus, § 30, § 6

nicht gleich (Test), § 66

nicht-programmierbare Operationen, § 56,  
§ 45, § 21

## O

---

oktal nach dezimal, § 42

Operationen, die  $R_9$  benutzen

## P

---

polar nach rechtwinklig, § 51, § 6

Präfix-Tasten, § 5

Programmierung, § 9

Programm-Modus, § 62

Programmschleifen, § 25, § 54, § 61, § 66

## Q

---

Quadrat, § 64

Quadratwurzel, § 64



## R

---

- rad, Winkelmodus, § 48, § 6
- rechtwinklig nach polar, § 51, § 6
- Redigieren (Korrigieren) eines Programms, § 9
- Return! (Zurück!), § 53, § 9
- Reziproker Wert von  $x$ , § 63
- Rückruf
  - Last X, § 38
  - Speicherregister  $R_n$ , § 49
- Rundung der Anzeige, § 24, § 4
- RUN-Modus, § 62

## S

---

- Schalter
  - Ein/Aus-Schalter, § 43
  - W/PRGM-RUN-Schalter, § 62
- Schiebe Stack nach unten, § 47
- Schiebe Stack nach oben, § 46
- Schrittweises Vorgehen, § 56, § 9
- Setze Flag, § 54
- Sinus, § 55, § 6
- Speichern in Register  $R_n$ , § 59, § 7
- Speicherregister-Arithmetik, § 59, § 7
- Starten eines Programms, § 52, § 15, § 53, § 31, § 9
- Start/Stop, § 52
- Stoppen eines Programms, § 52
- Subtraktion
  - Grad, Minuten, Sekunden, § 23
  - im Stack, § 11, § 2
  - der Speicherregister, § 59

## T

---

- Tangens, § 60, § 6
- Tausche  $x$  und  $y$ , § 65
- Testen
  - Flags, § 61
  - Verhältnis  $x$  zu  $y$ , § 66
  - Index  $r_8$ , § 25
- Trigonometrische Funktionen
  - Kosinus, Arkuskosinus, § 19, § 6
  - Sinus, Arkussinus, § 55, § 6
  - Tangens, Arkustangens, § 60, § 6
  - rechtwinklig nach/von polar, § 51, § 6

## U

---

- Überspringen von zwei Programmschritten, § 25, § 61, § 66
- Umkehrfunktionen
  - $10^x$ , § 37
  - Arkuskosinus, § 19
  - Arkussinus, § 55
  - Arkustangens, § 60
  - $e^x$ , § 36
  - Quadrat, § 64
  - subtrahiere D.MS, § 23

## V

---

- Vergleiche, § 66, § 25
- Verzweigung, § 31
- Vorzeichen von Zahl und Exponent, § 17, § 3
- Vorzeichenwechsel, § 17

## W

---

- Winkelmodus, § 6
- Wissenschaftliche Schreibweise
  - Anzeige, § 24, § 4
  - Eingabe von Zahlen, § 3, § 26, § 17



## **Z**

Zahleneingabe, Tasten für ..., § 12, § 13, § 17,  
§ 26

Ziffern, § 12

Zurück! (Return!), § 53, § 9







**172 mal Verkauf und Service in 65 Ländern**

**Hewlett-Packard GmbH/Vertrieb:**  
**6000 Frankfurt 56**, Berner Straße 117,  
Telefon (0611) 50 04-1

**3000 Hannover-Kleefeld**, Mellendorfer Straße 3,  
Telefon (0511) 55 06 26

**2000 Hamburg 1**, Wendenstraße 23,  
Telefon (040) 24 13 93

**8500 Nürnberg**, Hersbrückerstraße 42,  
Telefon (0911) 57 10 66

**8012 Ottobrunn**, Isar Center, Unterhachinger Straße 28,  
Telefon (089) 601 30 61/7

**7030 Böblingen**, Herrenbergerstraße 110,  
Telefon (07031) 66 72 87

**4000 Düsseldorf**, Vogelsanger Weg 38,  
Telefon (0211) 63 80 31/8

**1000 Berlin 3**, Keith Straße 2-4, Telefon (030) 24 90 86

**Für die Schweiz:** Hewlett-Packard (Schweiz) AG,  
Zürcherstraße 20, Postfach 64, 8952 Schlieren-Zürich,  
Telefon (01) 98 20 55

**Für Österreich/ Für sozialistische Staaten  
und UdSSR:** Hewlett-Packard Ges.m.b.H.,  
Handelskai 52/53, Postfach 7, A-1205 Wien, Österreich,  
Telefon (0222) 33 66 06 bis 09

**Europa-Zentrale:**  
Hewlett-Packard S.A., 7, rue du Bois-du-Lan, Postfach 85,  
CH-1217 Meyrin 2-Genf, Schweiz, Telefon (022) 41 54 00



Scan Copyright ©  
The Museum of HP Calculators  
[www.hpmuseum.org](http://www.hpmuseum.org)

Original content used with permission.

Thank you for supporting the Museum of HP  
Calculators by purchasing this Scan!

Please to not make copies of this scan or  
make it available on file sharing services.